Ex 6 Performance Analysis on Regression Techniques

September 11, 2023

[ ]:
```
Exp no: 6
Date: 04/09/2023
```

[ ]:
```
Bewin Felix R A
URK21CS1128
```

[ ]:
```
Aim:
    To execute the functionalities of Performance Analysis on Regression␣
 ↪Techniques using data science.
```

Description:

```
Machine learning has two types Supervised and Unsupervised learning. Supervised has Classifica
```

Regression: It predicts the continuous output variables based on the independent input variable. like the prediction of house prices based on different parameters like house age, distance from the main road, location, area, etc.

Linear regression is a type of supervised machine learning algorithm that computes the linear relationship between a dependent variable and one or more independent features. When the number of the independent feature, is 1 then it is known as Univariate Linear regression, and in the case of more than one feature, it is known as multivariate linear regression.

Linear regression is one of the most basic types of regression in machine learning. The linear regression model consists of a predictor variable and a dependent variable related linearly to each other. In case the data involves more than one independent variable, then linear regression is called multiple linear regression models.

Simple Linear Regression is a type of Regression algorithms that models the relationship between a dependent variable and a single independent variable. The relationship shown by a Simple Linear Regression model is linear or a sloped straight line, hence it is called Simple Linear Regression.

The key point in Simple Linear Regression is that the dependent variable must be a continuous/real value. However, the independent variable can be measured on continuous or categorical values.

Simple Linear regression algorithm has mainly two objectives:

Model the relationship between the two variables. Such as the relationship between Income and expenditure, experience and Salary, etc. Forecasting new observations. Such as Weather forecasting according to temperature, Revenue of a company according to the investments in a year, etc.

[1]:
```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from sklearn.metrics import mean_absolute_error,mean_squared_error,r2_score
from sklearn.linear_model import LinearRegression
import math
from sklearn.model_selection import train_test_split
```

[2]:
```
#1 a. Calculate the intercept and regression coefficients in y=b0+xb1
print('URK21CS1128')
sub=[1,2,3,4,5,6]
x=[43,21,25,42,57,59]
y=[99,65,79,75,87,81]
x=np.array(x)
y=np.array(y)
m_x=np.mean(x)
m_y=np.mean(y)
xx=x-m_x
yy=y-m_y
xy=xx*yy
xx2=xx*xx

b1=sum(xy)/sum(xx2)
b0=m_y-(b1*m_x)
print('Slope: ',b1)
print('Intercept: ',b0)
y_pred=b0+b1*x
```

```
URK21CS1128
Slope:  0.3852249832102082
Intercept:  65.1415715245131
```

[3]:
```
# b. Analyse the various performance metrics (Mean Squared Error, Mean Absolute␣
↪Error,Root Mean Squared Error, and R-Squared)
print('URK21CS1128')
print('MAE:',mean_absolute_error(y,y_pred))
print('MSE:',mean_squared_error(y,y_pred))
print('RMSE:',math.sqrt(mean_absolute_error(y,y_pred)))
print('R2 score:',r2_score(y,y_pred))
```

```
URK21CS1128
MAE: 7.173852697559885
MSE: 78.64374300425344
RMSE: 2.678404879319011
R2 score: 0.2806974725220722
```

[5]:
```
```

```
#2 Develop the linear regression model for the prediction of Graduate␣
 ↪Admissions from an Indian perspective in-terms of GRE Scores, LOR, CGPA␣
 ↪using the scikit-learn
df=pd.read_csv('Admission.csv',index_col=0)
x1=df['GRE Score']
x2=df['LOR ']
x3=df['CGPA']
y=df['Chance of Admit ']
# a. Divide the data into training (75%) and testing set (25%)
x1_train,x1_test,y_train,y_test=train_test_split(x1,y,test_size=0.
 ↪25,random_state=1)
x2_train,x2_test,y_train,y_test=train_test_split(x2,y,test_size=0.
 ↪25,random_state=1)
x3_train,x3_test,y_train,y_test=train_test_split(x3,y,test_size=0.
 ↪25,random_state=1)
```

[6]:
```
# b. Display the intercept and regression coefficients for the following cases
#       1. Analyse the impact of GRE scores to the Chance of Admit
#       2. Analyse the impact of LOR to the Chance of Admit
#       3. Analyse the impact of CGPA to the Chance of Admit
print('URK21CS1128')
x1_train=np.array(x1_train).reshape(-1,1)
y_train=np.array(y_train).reshape(-1,1)
model1=LinearRegression()
model1.fit(x1_train,y_train)
print('Impact of GRE Score on Chance of Admit')
print('Regression coefficient= ',model1.coef_)
print('Intercept= ',model1.intercept_,'\n')
x2_train=np.array(x2_train).reshape(-1,1)
y_train=np.array(y_train).reshape(-1,1)
model2=LinearRegression()
model2.fit(x2_train,y_train)
print('Impact of LOR on Chance of Admit')
print('Regression coefficient= ',model2.coef_)
print('Intercept= ',model2.intercept_,'\n')
x3_train=np.array(x3_train).reshape(-1,1)
y_train=np.array(y_train).reshape(-1,1)
model3=LinearRegression()
model3.fit(x3_train,y_train)
print('Impact of CGPA on Chance of Admit')
print('Regression coefficient= ',model3.coef_)
print('Intercept= ',model3.intercept_)
```

```
URK21CS1128
Impact of GRE Score on Chance of Admit
Regression coefficient=  [[0.01004167]]
Intercept=  [-2.45230421]
```

```
Impact of LOR on Chance of Admit
Regression coefficient=  [[0.09357372]]
Intercept=  [0.39662437]

Impact of CGPA on Chance of Admit
Regression coefficient=  [[0.20599061]]
Intercept=  [-1.04359588]
```

[7]:
```
# c. Predict the y value (y') for the testing set (x) and analyse the␣
 ↪performance metrics with the actual value (y) and predicted values (y') for␣
 ↪the above 3 cases
x1_test=np.array(x1_test).reshape(-1,1)
x2_test=np.array(x2_test).reshape(-1,1)
x3_test=np.array(x3_test).reshape(-1,1)
y1_p=model1.predict(x1_test)
y2_p=model2.predict(x2_test)
y3_p=model3.predict(x3_test)
y_test=np.array(y_test).reshape(-1,1)
```

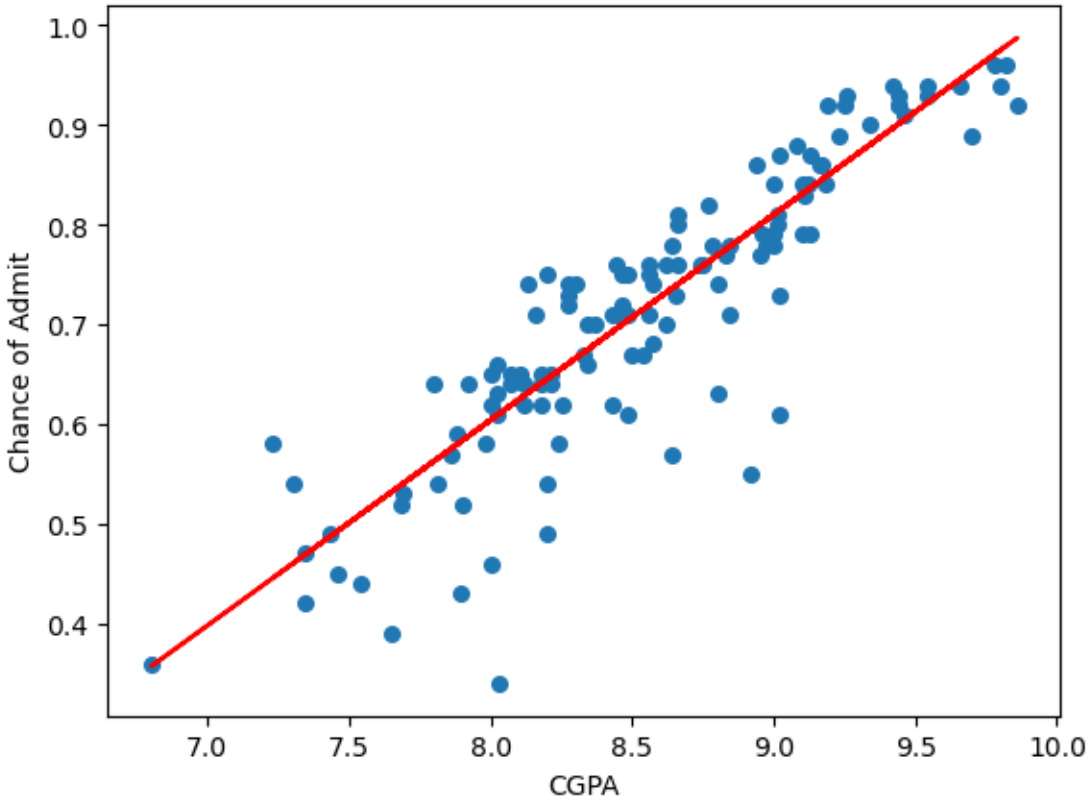
[8]:
```
print('URK21CS1128\n')
print('Impact of GRE Score on Chance of Admit')
print('MAE:',mean_absolute_error(y_test,y1_p))
print('MSE:',mean_squared_error(y_test,y1_p))
print('RMSE:',math.sqrt(mean_absolute_error(y_test,y1_p)))
print('R2 score:',r2_score(y_test,y1_p),'\n')
print('Impact of LOR on Chance of Admit')
print('MAE:',mean_absolute_error(y_test,y2_p))
print('MSE:',mean_squared_error(y_test,y2_p))
print('RMSE:',math.sqrt(mean_absolute_error(y_test,y2_p)))
print('R2 score:',r2_score(y_test,y2_p),'\n')
print('Impact of CGPA on Chance of Admit')
print('MAE:',mean_absolute_error(y_test,y3_p))
print('MSE:',mean_squared_error(y_test,y3_p))
print('RMSE:',math.sqrt(mean_absolute_error(y_test,y3_p)))
print('R2 score:',r2_score(y_test,y3_p))
```

```
URK21CS1128

Impact of GRE Score on Chance of Admit
MAE: 0.06264911695468035
MSE: 0.007625959038845016
RMSE: 0.25029805623432305
R2 score: 0.6260055317126936

Impact of LOR on Chance of Admit
MAE: 0.08129920544256705
MSE: 0.010893335796659088
```

```
RMSE: 0.2851301552669711
R2 score: 0.4657659045381354

Impact of CGPA on Chance of Admit
MAE: 0.04514566173687265
MSE: 0.004336070707235971
RMSE: 0.21247508497909268
R2 score: 0.7873491779396589
```

[9]:
```
# d. Identify the input parameter that has a greater impact in the prediction␣
 ↪of Graduate Admissions from an Indian perspective
print('URK21CS1128')
print('CGPA has the highest impact on the prediction of graduate admissions')
# e. Plot the regression line for the input parameter that has a greater impact␣
 ↪in the prediction of prediction of Graduate Admissions from an Indian␣
 ↪perspective
plt.scatter(x3_test,y_test,marker='o',s=30)
plt.plot(x3_test,y3_p,c='red')
plt.xlabel('CGPA')
plt.ylabel('Chance of Admit')
plt.show()
```

```
URK21CS1128
CGPA has the highest impact on the prediction of graduate admissions
```



[ ]:
```
Result:
    The functionalities of Performance Analysis on Regression Techniques using␣
 ↪data science were executed successfully
```